PLEASE AMEND THE SPECIFICATION AS FOLLOWS:

[0018] Referring now to **FIGURE 1**, there is illustrated a block diagram describing MPEG formatting of video data 305. The video data 305 comprises a series of frames 310. Each frame comprises two dimensional grids of luminance Y, chroma red Cr, and chroma blue Cb pixels 315. The two-dimensional grids are divided into 8x8 blocks 335, where four blocks 335 of luminance pixels Y are associated with a block 335 of chroma red Cr, and a block 335 of chroma blue Cb pixels. The four blocks of luminance pixels Y, the block of chroma red Cr, and the chroma blue Cb form a data structure known as a macroblock 337. The macroblock 337 also includes additional parameters, including motion vectors.

[0019] The macroblocks 337 representing a frame are grouped into different video packets 340. The video packet 340 includes the macroblocks 337 in the video packet 340, as well as additional parameters describing the video packet. Each of the video packets 340 forming the frame form the data portion of a picture structure 345. The picture 345 includes the video packets 340 as well as additional parameters. The pictures are then grouped together as a group of pictures 350. The group of pictures 350 also includes additional parameters. Groups of pictures 350 are then stored, forming what is known as a video elementary stream 355. The video elementary stream 355 is then packetized to form a packetized elementary sequence 360. Each packet is then associated with a transport header 365a, forming what are known as transport packets 365b. The transport packets 365b can be multiplexed with other transport packets 365b carrying other content, such as another video elementary stream 355 or an audio elementary stream. The multiplexed transport packets from what is known as a transport stream. The transport stream is transmitted over a communication medium for decoding and presentation.

[0020] The foregoing provides the data words 505(x)...505(n) as a set of 32 bit words starting from the last portion of 505(n) and proceeding sequentially to the first portion of 505(x). The bits forming the 32-bit words can be reversed with respect to one another, in any number of ways. For example, the MPEG video decoder 445 can include logic that reverses the 32 bits of each word. Alternatively, the DMA engine 510 can include additional circuitry 550 that causes the 32 bits of each word 611 to be provided to the MPEG video decoder 445 in the reverse order.

[0021] Referring now to **FIGURE 2**, there is illustrated a block diagram of an exemplary decoder 400 for decoding compressed video data, configured in accordance with an embodiment of the present invention. A processor, that may include a CPU 490, reads a stream of transport packets 365b (a transport stream) into a transport stream buffer 432 within an SDRAM 430.

[0022] The data is output from the transport stream presentation buffer 432 and is then passed to a data transport processor 435. The data transport processor then demultiplexes the MPEG transport stream into its PES constituents and passes the audio transport stream to an audio decoder 460 or SPDIF Generator 470 and the video transport stream to a video transport processor 440.

[0040] The fetched data words 505 are stored (720) in the local buffer 610. After the data words are stored in the local buffer 610, the state logic machine 605 causes the contents of the local buffer 610 to be provided (725) via port 560 to the MPEG video decoder 445 beginning with word 611(127), and proceeding sequentially to word 611(0). After the contents of the local buffer 610, e.g., words 611(127)...611(0), are provided to the MPEG video decoder 445, a determination (730) is made whether the first data word, data word 505(x) has been provided to the MPEG video decoder 445. If the first data word has not been provided to the MPEG video decoder at 730, the next batch if prepared at 735, and 705-730 are

repeated. If the first data word has been provided to the MPEG video decoder at 730, the process is completed.